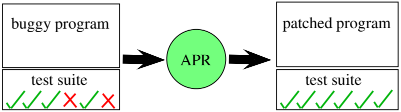
Do Automated Program Repair Techniques Repair Hard and Important Bugs?

Manish Motwani

Sandhya Sankarnarayanan René Just Yuriy Brun

University of Massachusetts Amherst

Automatic Program Repair: An Active Research Area



buggy program

test suite

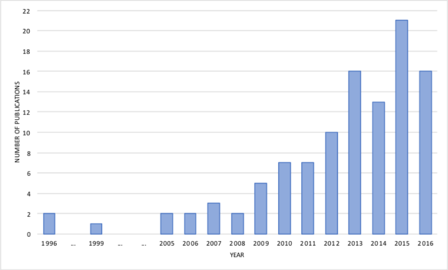
✓✓✓✗✓✗

APR

patched program

test suite

✓✓✓✓✓✓

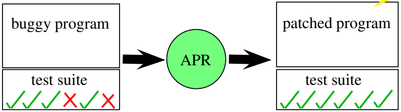


Automated program repair publications per year [1]

[1] Gazzola, Micucci, and Mariani. Automatic software repair: A survey. IEEE TSE 2017.

Automatic Program Repair: An Active Research Area

Is the patched program correct?



buggy program

test suite

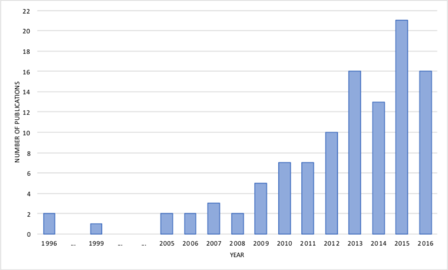
✓✓✓✗✓✗

APR

patched program

test suite

✓✓✓✓✓✓



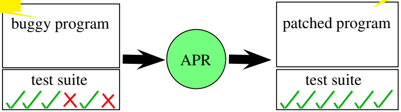
Automated program repair publications per year [1]

[1] Gazzola, Micucci, and Mariani. Automatic software repair: A survey. IEEE TSE 2017.

Automatic Program Repair: An Active Research Area

Is the bug hard to fix?

Is the patched program correct?



buggy program

test suite

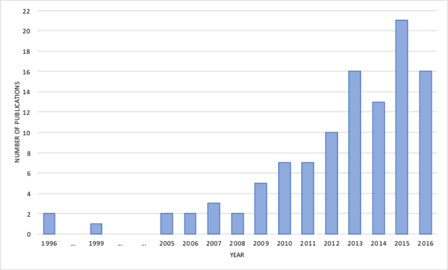
✓✓✓✗✓✗

APR

patched program

test suite

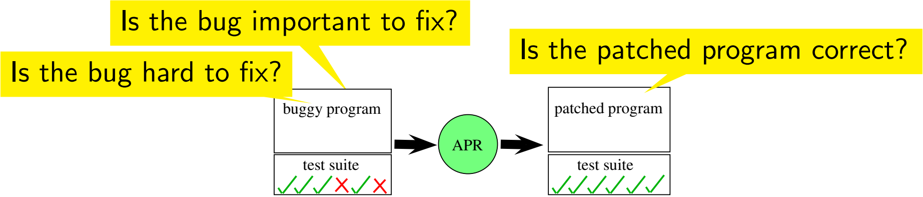
✓✓✓✓✓✓



Automated program repair publications per year [1]

[1] Gazzola, Micucci, and Mariani. Automatic software repair: A survey. IEEE TSE 2017.

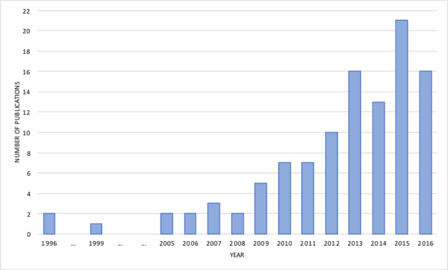
Automatic Program Repair: An Active Research Area



Is the bug important to fix?

Is the bug hard to fix?

Is the patched program correct?



Automated program repair publications per year [1]

[1] Gazzola, Micucci, and Mariani. Automatic software repair: A survey. IEEE TSE 2017.

Motivation

Prior evaluations of automated repair have focused on:

- Fraction of defects repaired [1,2]
- Computational resources required to repair defects [3,4]
- Correctness and quality of generated patches [5,6,7]
- Patch maintainability [8]
- Repair acceptability [9,10]

[1] Ke et al. Repairing programs with semantic code search. ASE. 2015.
[2] Qi et al. An analysis of patch plausibility and correctness for G&V patch generation systems. ISSTA. 2015.
[3] Le Goues et al. The ManyBugs and IntroClass benchmarks for automated repair of C programs. TSE. 2015
[4] Weimer et al. Leveraging program equivalence for adaptive program repair: models and first results. ASE. 2013
[5] (DBGBench) Boehme, et al. Where is the bug and how is it fixed? an experiment with practitioners. FSE. 2017.
[6] Smith et al. Is the cure worse than the disease? Overfitting in automated program repair. FSE. 2015.
[7] Pei et al. Automated fixing of programs with contracts. TSE. 2014.
[8] Fry et al. A human study of patch maintainability. ISSTA. 2012.
[9] Durieux et al. Automatic repair of real bugs: An experience report on the Defects4J dataset. 2015.
[10] Kim et al. Automatic patch generation learned from human-written patches. ICSE. 2013.

Motivation

YetAnotherFix
fixes 60% of the defects

Defect-1	patched
Defect-2	patched
Defect-3	not patched
Defect-4	patched
Defect-5	patched
Defect-6	not patched
Defect-7	patched
Defect-8	patched
Defect-9	not patched
Defect-10	not patched

ThisNeverEndsFix
fixes 30% of the defects

Defect-1	not patched
Defect-2	not patched
Defect-3	patched
Defect-4	not patched
Defect-5	not patched
Defect-6	not patched
Defect-7	not patched
Defect-8	not patched
Defect-9	patched
Defect-10	patched

Which automated program repair technique is better?

Motivation

YetAnotherFix
fixes 60% of the defects

Defect-1	patched
Defect-2	patched
Defect-3	**not patched**
Defect-4	patched
Defect-5	patched
Defect-6	not patched
Defect-7	patched
Defect-8	patched
Defect-9	not patched
Defect-10	**not patched**

Hard to fix defects

ThisNeverEndsFix
fixes 30% of the defects

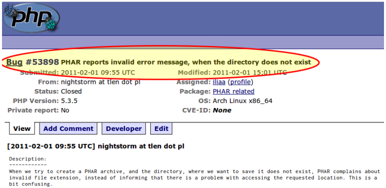
Defect-1	not patched
Defect-2	not patched
Defect-3	**patched**
Defect-4	not patched
Defect-5	not patched
Defect-6	not patched
Defect-7	not patched
Defect-8	not patched
Defect-9	patched
Defect-10	**patched**

Which automated program repair technique is better?

How about now?

Which is harder to fix?

Invalid error message



Bug #53898 PHAR reports invalid error message, when the directory does not exist

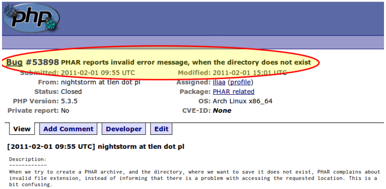
Submitted:	2011-02-01 09:55 UTC	**Modified:**	2011-02-01 15:01 UTC
From:	nightstorm at tlen dot pl	**Assigned:**	iliaa (profile)
Status:	Closed	**Package:**	PHAR related
PHP Version:	5.3.5	**OS:**	Arch Linux x86_64
Private report:	No	**CVE-ID:**	***None***

[2011-02-01 09:55 UTC] nightstorm at tlen dot pl

```
Description:
------------
When we try to create a PHAR archive, and the directory, where we want to save it does not exist, PHAR complains about
invalid file extension, instead of informing that there is a problem with accessing the requested location. This is a
bit confusing.
```

Which is harder to fix?

Invalid error message



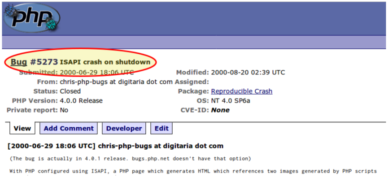
Bug #53898 **PHAR reports invalid error message, when the directory does not exist**

Submitted: 2011-02-01 09:55 UTC
Modified: 2011-02-01 15:01 UTC
From: nightstorm at tlen dot pl
Assigned: iliaa (profile)
Status: Closed
Package: PHAR related
PHP Version: 5.3.5
OS: Arch Linux x86_64
Private report: No
CVE-ID: ***None***

[2011-02-01 09:55 UTC] nightstorm at tlen dot pl

```
Description:
------------
When we try to create a PHAR archive, and the directory, where we want to save it does not exist, PHAR complains about
invalid file extension, instead of informing that there is a problem with accessing the requested location. This is a
bit confusing.
```

Invalid memory access (Application crash)



Bug #5273 **ISAPI crash on shutdown**

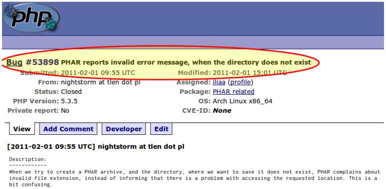
Submitted: 2000-06-29 18:06 UTC
Modified: 2000-08-20 02:39 UTC
From: chris-php-bugs at digitaria dot com
Assigned:
Status: Closed
Package: Reproducible Crash
PHP Version: 4.0.0 Release
OS: NT 4.0 SP6a
Private report: No
CVE-ID: ***None***

[2000-06-29 18:06 UTC] chris-php-bugs at digitaria dot com

```
(The bug is actually in 4.0.1 release. bugs.php.net doesn't have that option)

With PHP configured using ISAPI, a PHP page which generates HTML which references two images generated by PHP scripts
```

Which is harder to fix? Which is more important to fix?

Invalid error message



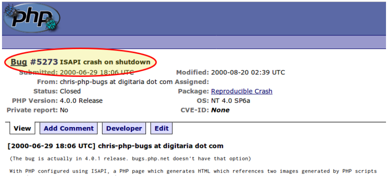
Bug #53898 PHAR reports invalid error message, when the directory does not exist

Submitted: 2011-02-01 09:55 UTC
Modified: 2011-02-01 15:01 UTC
From: nightstorm at tlen dot pl
Assigned: iliaa (profile)
Status: Closed
Package: PHAR related
PHP Version: 5.3.5
OS: Arch Linux x86_64
Private report: No
CVE-ID: *None*

[2011-02-01 09:55 UTC] nightstorm at tlen dot pl

```
Description:
------------
When we try to create a PHAR archive, and the directory, where we want to save it does not exist, PHAR complains about
invalid file extension, instead of informing that there is a problem with accessing the requested location. This is a
bit confusing.
```

Invalid memory access
(Application crash)



Bug #5273 ISAPI crash on shutdown

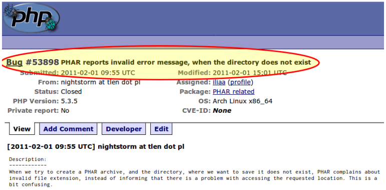
Submitted: 2000-06-29 18:06 UTC
Modified: 2000-08-20 02:39 UTC
From: chris-php-bugs at digitaria dot com
Assigned:
Status: Closed
Package: Reproducible Crash
PHP Version: 4.0.0 Release
OS: NT 4.0 SP6a
Private report: No
CVE-ID: *None*

[2000-06-29 18:06 UTC] chris-php-bugs at digitaria dot com

```
(The bug is actually in 4.0.1 release. bugs.php.net doesn't have that option)

With PHP configured using ISAPI, a PHP page which generates HTML which references two images generated by PHP scripts
```

Which is harder to fix? Which is more important to fix?

Invalid error message



Bug #53898 PHAR reports invalid error message, when the directory does not exist

Submitted: 2011-02-01 09:55 UTC
From: nightstorm at tlen dot pl
Status: Closed
PHP Version: 5.3.5
Private report: No

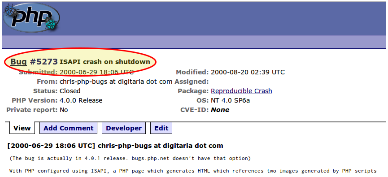
Modified: 2011-02-01 15:01 UTC
Assigned: iliaa (profile)
Package: PHAR related
OS: Arch Linux x86_64
CVE-ID: ***None***

[2011-02-01 09:55 UTC] nightstorm at tlen dot pl

```
Description:
------------
When we try to create a PHAR archive, and the directory, where we want to save it does not exist, PHAR complains about
invalid file extension, instead of informing that there is a problem with accessing the requested location. This is a
bit confusing.
```

Easy and less important

Invalid memory access
(Application crash)



Bug #5273 ISAPI crash on shutdown

Submitted: 2000-06-29 18:06 UTC
From: chris-php-bugs at digitaria dot com
Status: Closed
PHP Version: 4.0.0 Release
Private report: No

Modified: 2000-08-20 02:39 UTC
Assigned:
Package: Reproducible Crash
OS: NT 4.0 SP6a
CVE-ID: ***None***

[2000-06-29 18:06 UTC] chris-php-bugs at digitaria dot com

```
(The bug is actually in 4.0.1 release. bugs.php.net doesn't have that option)

With PHP configured using ISAPI, a PHP page which generates HTML which references two images generated by PHP scripts
```

Hard and more important

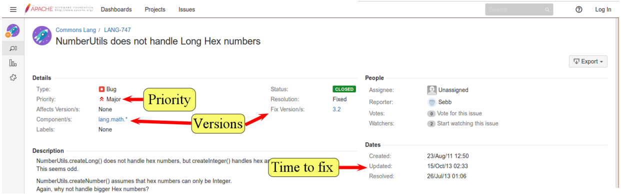
How do we measure hardness and importance of a defect?

Goals of this study

A methodology for measuring a defect's hardness and importance.

An evaluation of whether automated program repair techniques repair hard and important defects.

Measuring hardness and importance of a defect



Commons Lang / LANG-747

NumberUtils does not handle Long Hex numbers

Details

Type:	Bug	Status:	CLOSED
Priority:	Major	Resolution:	Fixed
Affects Version/s:	None	Fix Version/s:	3.2
Component/s:	lang.math.*		
Labels:	None		

People

Assignee:	Unassigned
Reporter:	Sebb
Votes:	0 Vote for this issue
Watchers:	2 Start watching this issue

Description

NumberUtils.createLong() does not handle hex numbers, but createInteger() handles hex ar
This seems odd.

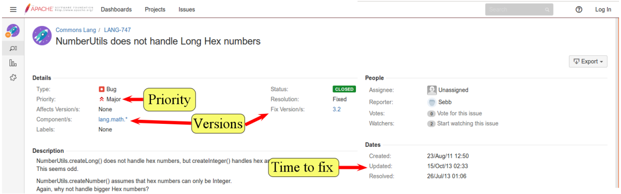
NumberUtils.createNumber() assumes that hex numbers can only be Integer.
Again, why not handle bigger Hex numbers?

Dates

Created:	23/Aug/11 12:50
Updated:	15/Oct/13 02:33
Resolved:	26/Jul/13 01:06

bug report

Measuring hardness and importance of a defect



Commons Lang / LANG-747

NumberUtils does not handle Long Hex numbers

Details

Type:	Bug
Priority:	Major
Affects Version/s:	None
Component/s:	lang.math.*
Labels:	None
Status:	CLOSED
Resolution:	Fixed
Fix Version/s:	3.2

People

Assignee:	Unassigned
Reporter:	Sebb
Votes:	0 Vote for this issue
Watchers:	2 Start watching this issue

Dates

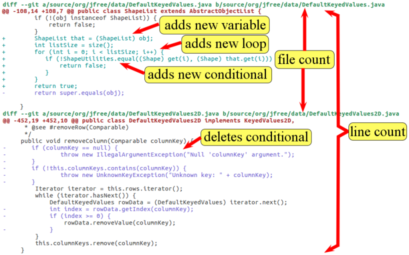
Created:	23/Aug/11 12:50
Updated:	15/Oct/13 02:33
Resolved:	26/Jul/13 01:06

Description

NumberUtils.createLong() does not handle hex numbers, but createInteger() handles hex ar
This seems odd.

NumberUtils.createNumber() assumes that hex numbers can only be Integer.
Again, why not handle bigger Hex numbers?

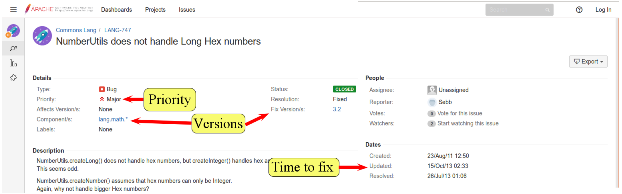
bug report



```
diff --git a/source/org/jfree/data/DefaultKeyedValues.java b/source/org/jfree/data/DefaultKeyedValues.java
@@ -108,14 +108,7 @@ public class ShapeList extends AbstractObjectList {
         if (!(obj instanceof ShapeList)) {
             return false;
         }
+        ShapeList that = (ShapeList) obj;
+        int listSize = size();
+        for (int i = 0; i < listSize; i++) {
+           if (!ShapeUtilities.equal((Shape) get(i), (Shape) that.get(i)))
+               return false;
+           }
+        }
+        return true;
-        return super.equals(obj);

    }
diff --git a/source/org/jfree/data/DefaultKeyedValues2D.java b/source/org/jfree/data/DefaultKeyedValues2D.java
@@ -452,19 +452,10 @@ public class DefaultKeyedValues2D implements KeyedValues2D,
      * @see #removeRow(Comparable)
      */
    public void removeColumn(Comparable columnKey) {
-       if (columnKey == null) {
-               throw new IllegalArgumentException("Null 'columnKey' argument.");
-       }
-       if (!this.columnKeys.contains(columnKey)) {
-               throw new UnknownKeyException("Unknown key: " + columnKey);
-       }
        Iterator iterator = this.rows.iterator();
        while (iterator.hasNext()) {
            DefaultKeyedValues rowData = (DefaultKeyedValues) iterator.next();
-           int index = rowData.getIndex(columnKey);
-           if (index >= 0) {
                rowData.removeValue(columnKey);
-           }
        }
        this.columnKeys.remove(columnKey);
    }
```

Developer-written patch

Measuring hardness and importance of a defect



Commons Lang / LANG-747

NumberUtils does not handle Long Hex numbers

Details

Type:	Bug	Status:	CLOSED
Priority:	Major	Resolution:	Fixed
Affects Version/s:	None	Fix Version/s:	3.2
Component/s:	lang.math.*		
Labels:	None		

People

Assignee:	Unassigned
Reporter:	Sebb
Votes:	0 Vote for this issue
Watchers:	2 Start watching this issue

Dates

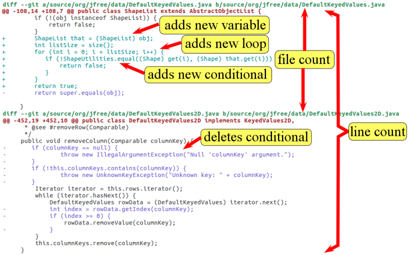
Created:	23/Aug/11 12:50
Updated:	15/Oct/13 02:33
Resolved:	26/Jul/13 01:06

Description

NumberUtils.createLong() does not handle hex numbers, but createInteger() handles hex ar
This seems odd.

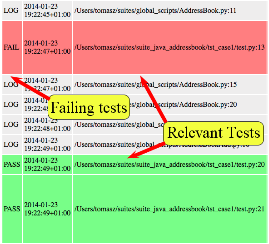
NumberUtils.createNumber() assumes that hex numbers can only be Integer.
Again, why not handle bigger Hex numbers?

bug report



```
diff --git a/source/org/jfree/data/DefaultKeyedValues.java b/source/org/jfree/data/DefaultKeyedValues.java
@@ -108,14 +108,7 @@ public class ShapeList extends AbstractObjectList {
         if (!(obj instanceof ShapeList)) {
             return false;
         }
+        ShapeList that = (ShapeList) obj;
+        int listSize = size();
+        for (int i = 0; i < listSize; i++) {
+           if (!ShapeUtilities.equal((Shape) get(i), (Shape) that.get(i)))
+               return false;
+           }
+        }
+        return true;
-        return super.equals(obj);

     }
diff --git a/source/org/jfree/data/DefaultKeyedValues2D.java b/source/org/jfree/data/DefaultKeyedValues2D.java
@@ -452,19 +452,10 @@ public class DefaultKeyedValues2D implements KeyedValues2D,
      * @see #removeRow(Comparable)
      */
     public void removeColumn(Comparable columnKey) {
-       if (columnKey == null) {
-               throw new IllegalArgumentException("Null 'columnKey' argument.");
-       }
-       if (!this.columnKeys.contains(columnKey)) {
-               throw new UnknownKeyException("Unknown key: " + columnKey);
-       }
        Iterator iterator = this.rows.iterator();
        while (iterator.hasNext()) {
            DefaultKeyedValues rowData = (DefaultKeyedValues) iterator.next();
-           int index = rowData.getIndex(columnKey);
-           if (index >= 0) {
               rowData.removeValue(columnKey);
-           }
        }
        this.columnKeys.remove(columnKey);
     }
```

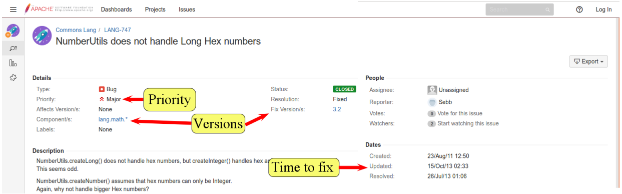
Developer-written patch



LOG	2014-01-23 19:22:45+01:00	/Users/tomasz/suites/global_scripts/AddressBook.py:11
FAIL	2014-01-23 19:22:47+01:00	/Users/tomasz/suites/suite_java_addressbook/tst_case1/test.py:13
LOG	2014-01-23 19:22:47+01:00	/Users/tomasz/suites/global_scripts/AddressBook.py:15
LOG	2014-01 19:22:48	s/global_scripts/AddressBook.py:20
LOG	2014-01-23 19:22:48+01:00	/Users/tomasz/suites/global_sc
LOG	2014-01-23 19:22:49+01:00	/Users/tomasz/suites/global
PASS	2014-01-23 19:22:49+01:00	/Users/tomasz/suites/suite_java_addressbook/tst_case1/test.py:20
PASS	2014-01-23 19:22:49+01:00	/Users/tomasz/suites/suite_java_addressbook/tst_case1/test.py:21

Test-suite

Measuring hardness and importance of a defect



Commons Lang / LANG-747

NumberUtils does not handle Long Hex numbers

Details

Type:	Bug
Priority:	Major
Affects Version/s:	None
Component/s:	lang.math.*
Labels:	None
Status:	CLOSED
Resolution:	Fixed
Fix Version/s:	3.2

People

Assignee:	Unassigned
Reporter:	Sebb
Votes:	0 Vote for this issue
Watchers:	2 Start watching this issue

Dates

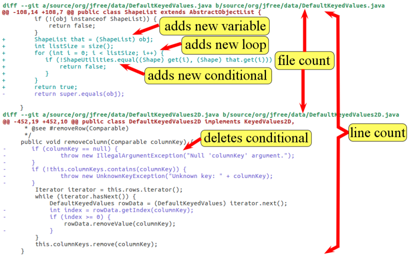
Created:	23/Aug/11 12:50
Updated:	15/Oct/13 02:33
Resolved:	26/Jul/13 01:06

Description

NumberUtils.createLong() does not handle hex numbers, but createInteger() handles hex a[illegible]
This seems odd.

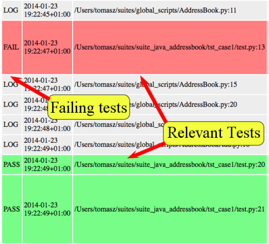
NumberUtils.createNumber() assumes that hex numbers can only be Integer.
Again, why not handle bigger Hex numbers?

bug report



```
diff --git a/source/org/jfree/data/DefaultKeyedValues.java b/source/org/jfree/data/DefaultKeyedValues.java
@@ -108,14 +108,7 @@ public class ShapeList extends AbstractObjectList {
         if (!(obj instanceof ShapeList)) {
             return false;
         }
+        ShapeList that = (ShapeList) obj;
+        int listSize = size();
+        for (int i = 0; i < listSize; i++) {
+           if (!ShapeUtilities.equal((Shape) get(i), (Shape) that.get(i)))
+               return false;
+           }
+        }
+        return true;
-        return super.equals(obj);

     }
diff --git a/source/org/jfree/data/DefaultKeyedValues2D.java b/source/org/jfree/data/DefaultKeyedValues2D.java
@@ -452,19 +452,10 @@ public class DefaultKeyedValues2D implements KeyedValues2D,
      * @see #removeRow(Comparable)
      */
     public void removeColumn(Comparable columnKey) {
-       if (columnKey == null) {
-               throw new IllegalArgumentException("Null 'columnKey' argument.");
-       }
-       if (!this.columnKeys.contains(columnKey)) {
-               throw new UnknownKeyException("Unknown key: " + columnKey);
-       }
        Iterator iterator = this.rows.iterator();
        while (iterator.hasNext()) {
            DefaultKeyedValues rowData = (DefaultKeyedValues) iterator.next();
-           int index = rowData.getIndex(columnKey);
-           if (index >= 0) {
               rowData.removeValue(columnKey);
-           }
        }
        this.columnKeys.remove(columnKey);
     }
```

Developer-written patch



LOG	2014-01-23 19:22:45+01:00	/Users/tomasz/suites/global_scripts/AddressBook.py:11
FAIL	2014-01-23 19:22:47+01:00	/Users/tomasz/suites/suite_java_addressbook/tst_case1/test.py:13
LOG	2014-01-23 [illegible]:47+01:00	/Users/tomasz/suites/global_scripts/AddressBook.py:15
LOG	2014-01[illegible] 19:22:4[illegible]	[illegible]s/global_scripts/AddressBook.py:20
LOG	2014-01-23 19:22:48+01:00	/Users/tomasz/suites/global_sc[illegible]
LOG	2014-01-23 19:22:49+01:00	/Users/tomasz/suites/global[illegible]
PASS	2014-01-23 19:22:49+01:00	/Users/tomasz/suites/suite_java_addressbook/tst_case1/test.py:20
PASS	2014-01-23 19:22:49+01:00	/Users/tomasz/suites/suite_java_addressbook/tst_case1/test.py:21

Test-suite

Other parameters may also exist.

Measuring hardness and importance of a defect

Analyzed 8 popular **bug-tracking systems**

IBM. Rational. ClearQuest

hp Quality Center

mantis BUG TRACKER

REDMINE flexible project management

JIRA

Bugzilla

FogBugz

trac Integrated SCM & Project Management

Analyzed 3 popular **open-source code repositories**

GitHub

sourceforge

Google code

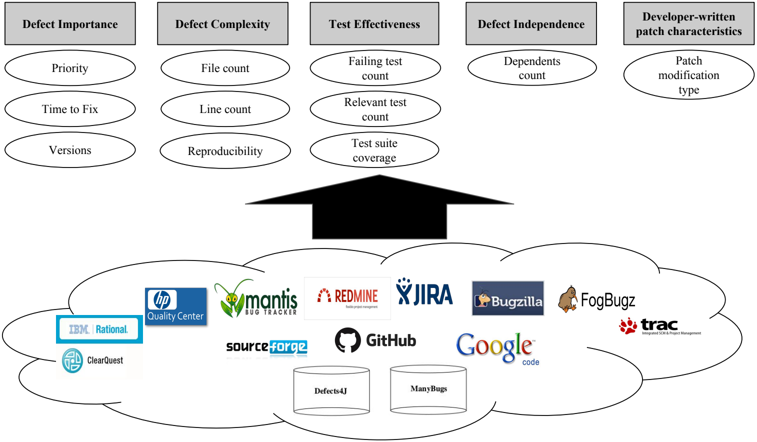
Analyzed 2 **defect benchmarks**

Defects4J

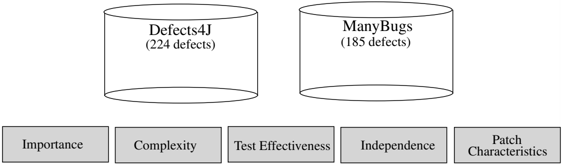
ManyBugs

Measuring hardness and importance of a defect

5 defect characteristics defined in terms of 11 **abstract parameters**

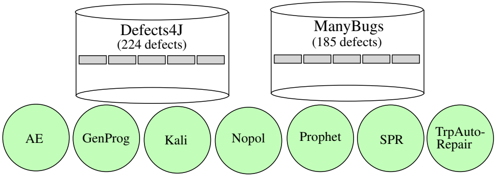


Evaluating repair techniques along new dimensions



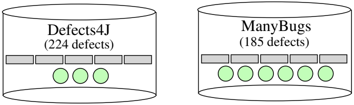
- 2 defect benchmarks: Defects4J and ManyBugs
- Semi-automatically annotated 409 defects with:
    - 5 defects characteristics defined using 11 abstract parameters.

Evaluating repair techniques along new dimensions



- 2 defect benchmarks: Defects4J and ManyBugs
- Semi-automatically annotated 409 defects with:
  - 5 defects characteristics defined using 11 abstract parameters.
  - Existing repairability and repair quality results of 7 automated repair techniques.

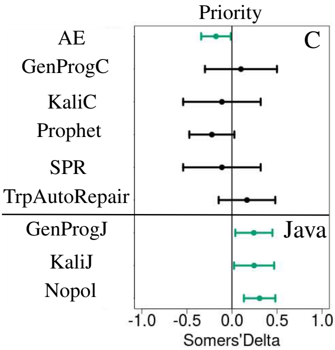
Evaluating repair techniques along new dimensions



- 2 defect benchmarks: Defects4J and ManyBugs
- Semi-automatically annotated 409 defects with:
  - 5 defects characteristics defined using 11 abstract parameters.
  - Existing repairability and repair quality results of 7 automated repair techniques.
- Identify if repairability of a repair technique correlates (Somer's Delta $\in [-1, 1]$) with each abstract parameter.

Do repair techniques repair important defects?

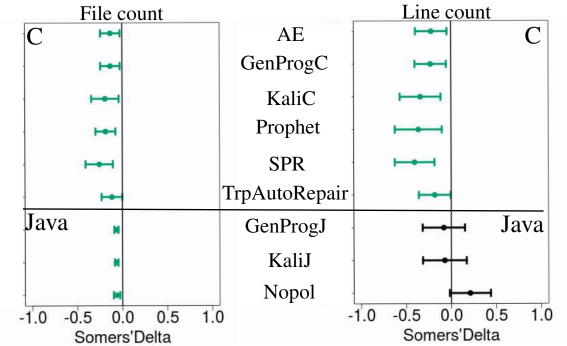
Importance | Complexity | Test Effectiveness | Patch Characteristics



Java repair techniques are more likely to repair defects that are important for developers.

Do repair techniques repair hard defects?

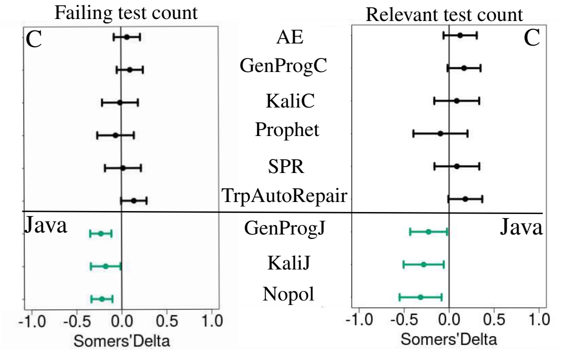
Importance | Complexity | Test Effectiveness | Patch Characteristics



C repair techniques are less likely to repair defects that required developers to write more code.

Do repair techniques repair defects with effective test suites?

Importance | Complexity | Test Effectiveness | Patch Characteristics

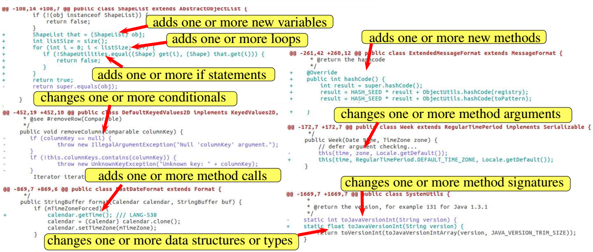


Java repair techniques are less likely to repair defects with effective test suites.

What patch modification types are challenging for automated repair?

Importance | Complexity | Test Effectiveness | **Patch Characteristics**

9 Patch modification types [1]



adds one or more new variables

adds one or more loops

adds one or more if statements

changes one or more conditionals

```
@@ -108,14 +108,7 @@ public class ShapeList extends AbstractObjectList {
         if (!(obj instanceof ShapeList)) {
             return false;
         }
+        ShapeList that = (ShapeList) obj;
+        int listSize = size();
+        for (int i = 0; i < listSize; i++) {
+           if (!ShapeUtilities.equal((Shape) get(i), (Shape) that.get(i))) {
+               return false;
+           }
+        }
+        return true;
-        return super.equals(obj);

     }
```

```
@@ -452,19 +452,10 @@ public class DefaultKeyedValues2D implements KeyedValues2D,
      * @see #removeRow(Comparable)
      */
     public void removeColumn(Comparable columnKey) {
-        if (columnKey == null) {
-              throw new IllegalArgumentException("Null 'columnKey' argument.");
-        }
-        if (!this.columnKeys.contains(columnKey)) {
-              throw new UnknownKeyException("Unknown key: " + columnKey);
-        }
         Iterator iterat
```

adds one or more method calls

```
@@ -869,7 +869,6 @@ public class FastDateFormat extends Format {
      */
     public StringBuffer format(Calendar calendar, StringBuffer buf) {
         if (mTimeZoneForced) {
+             calendar.getTime(); /// LANG-538
             calendar = (Calendar) calendar.clone();
             calendar.setTimeZone(mTimeZone);
         }
```

changes one or more data structures or types

adds one or more new methods

```
@@ -261,42 +260,12 @@ public class ExtendedMessageFormat extends MessageFormat {
      * @return the hashcode
      */
+    @Override
+    public int hashCode() {
+        int result = super.hashCode();
+        result = HASH_SEED * result + ObjectUtils.hashCode(registry);
+        result = HASH_SEED * result + ObjectUtils.hashCode(toPattern);

+    }
```

changes one or more method arguments

```
@@ -172,7 +172,7 @@ public class Week extends RegularTimePeriod implements Serializable {
      */
     public Week(Date time, TimeZone zone) {
         // defer argument checking...
-        this(time, zone, Locale.getDefault());
+        this(time, RegularTimePeriod.DEFAULT_TIME_ZONE, Locale.getDefault());
     }
```

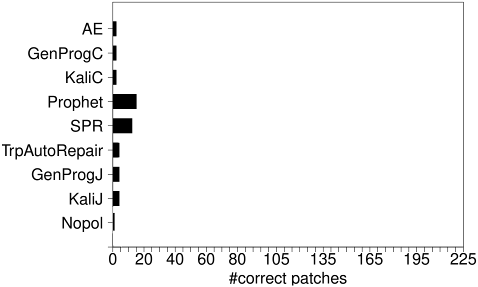
changes one or more method signatures

```
@@ -1669,7 +1669,7 @@ public class SystemUtils {
      *
      * @return the version, for example 131 for Java 1.3.1
      */
-    static int toJavaVersionInt(String version) {
+    static float toJavaVersionInt(String version) {
         return toVersionInt(toJavaVersionIntArray(version, JAVA_VERSION_TRIM_SIZE));
     }
```

Defects that required developers to **add loops or a new method call, or change a method signature** are challenging for automated repair techniques to patch.

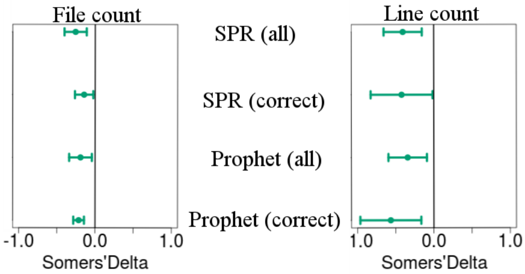
[1] Le Goues et al. The ManyBugs and IntroClass benchmarks for automated repair of C programs. IEEE TSE 2015.

What about correct patches?



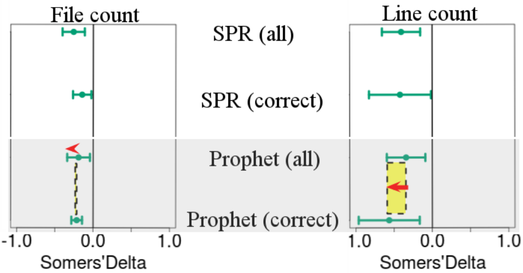
Only Prophet (15) and SPR (13) generate sufficient number of correct patches.

What about correct patches?



Prophet is less likely to produce patches for more complex defects, and even less likely to produce correct patches for the same defects.

What about correct patches?

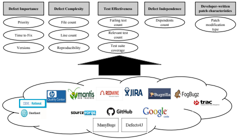


Prophet is less likely to produce patches for more complex defects, and even less likely to produce correct patches for the same defects.

Contributions

Methodology to measure importance and hardness of a defect.

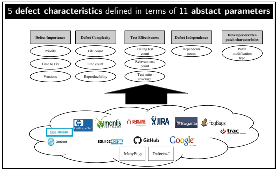
5 **defect characteristics** defined in terms of 11 **abstact parameters**



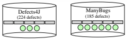
Contributions

Methodology to measure importance and hardness of a defect.

Methodology to evaluate automated program repair techniques along new dimensions.



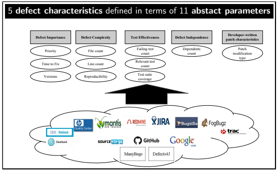
5 **defect characteristics** defined in terms of 11 **abstact parameters**



- 2 defect benchmarks: Defects4J and ManyBugs
- Annotated 409 defects with:
  - 5 defects characteristics defined using 11 abstract parameters.
  - Existing repairability and repair quality results of 7 automated repair techniques.
- Identify if repairability of a repair technique correlates (Somer's Delta $\in [-1, 1]$) with each abstract parameter.

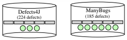
Contributions

Methodology to measure importance and hardness of a defect.



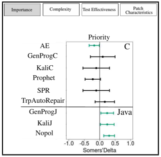
5 **defect characteristics** defined in terms of 11 **abstact parameters**

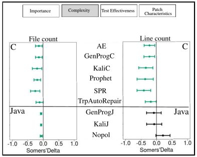
Methodology to evaluate automated program repair techniques along new dimensions.

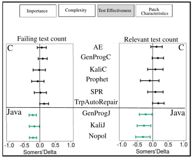


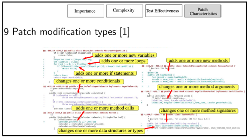
- 2 defect benchmarks: Defects4J and ManyBugs
- Annotated 409 defects with:
  - 5 defects characteristics defined using 11 abstract parameters.
  - Existing repairability and repair quality results of 7 automated repair techniques.
- Identify if repairability of a repair technique correlates (Somer's Delta $\in [-1, 1]$) with each abstract parameter.

Evaluation of 7 automated program repair techniques on 409 real-world defects.







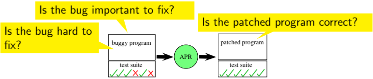


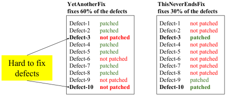
9 Patch modification types [1]

Importance
Complexity
Test Effectiveness
Patch Characteristics
adds one or more new variables
adds one or more loops
adds one or more new methods
adds one or more if statements
changes one or more conditionals
changes one or more method arguments
adds one or more method calls
changes one or more method signatures
changes one or more data structures or types

Recommendations

Repair research should evaluate if new techniques repair hard and important defects.




Hard to fix defects

YetAnotherFix
fixes 60% of the defects

Defect-1	patched
Defect-2	patched
Defect-3	**not patched**
Defect-4	patched
Defect-5	patched
Defect-6	not patched
Defect-7	patched
Defect-8	patched
Defect-9	not patched
Defect-10	**not patched**

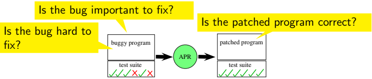
ThisNeverEndsFix
fixes 30% of the defects

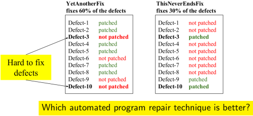
Defect-1	not patched
Defect-2	not patched
Defect-3	**patched**
Defect-4	not patched
Defect-5	not patched
Defect-6	not patched
Defect-7	not patched
Defect-8	not patched
Defect-9	patched
Defect-10	**patched**

Which automated program repair technique is better?

Recommendations

Repair research should evaluate if new techniques repair hard and important defects.



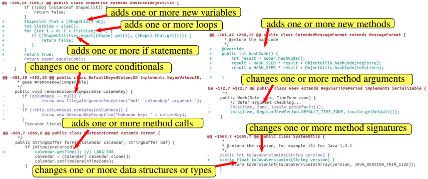


YetAnotherFix fixes 60% of the defects		ThisNeverEndsFix fixes 30% of the defects	
Defect-1	patched	Defect-1	not patched
Defect-2	patched	Defect-2	not patched
Defect-3	**not patched**	**Defect-3**	**patched**
Defect-4	patched	Defect-4	not patched
Defect-5	patched	Defect-5	not patched
Defect-6	not patched	Defect-6	not patched
Defect-7	patched	Defect-7	not patched
Defect-8	patched	Defect-8	not patched
Defect-9	not patched	Defect-9	patched
Defect-10	**not patched**	**Defect-10**	**patched**

Hard to fix defects

Which automated program repair technique is better?

Repair research should target defects that existing techniques have missed.

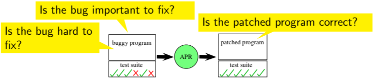

adds one or more new variables
adds one or more loops
adds one or more if statements
changes one or more conditionals
adds one or more method calls
changes one or more data structures or types
adds one or more new methods
changes one or more method arguments
changes one or more method signatures

```
@@ -108,14 +108,7 @@ public class ShapeList extends AbstractObjectList {
         if (!(obj instanceof ShapeList)) {
             return false;
         }
+        ShapeList that = (ShapeList) obj;
+        int listSize = size();
+        for (int i = 0; i < listSize; i++) {
+            if (!ShapeUtilities.equal((Shape) get(i), (Shape) that.get(i))) {
+                return false;
+            }
+        }
+        return true;
-        return super.equals(obj);

     }

@@ -452,19 +452,10 @@ public class DefaultKeyedValues2D implements KeyedValues2D,
      * @see #removeRow(Comparable)
      */
     public void removeColumn(Comparable columnKey) {
-        if (columnKey == null) {
-            throw new IllegalArgumentException("Null 'columnKey' argument.");
-        }
-        if (!this.columnKeys.contains(columnKey)) {
-            throw new UnknownKeyException("Unknown key: " + columnKey);
-        }
         Iterator iterat

@@ -869,7 +869,6 @@ public class FastDateFormat extends Format {
      */
     public StringBuffer format(Calendar calendar, StringBuffer buf) {
         if (mTimeZoneForced) {
+            calendar.getTime(); /// LANG-538
             calendar = (Calendar) calendar.clone();
             calendar.setTimeZone(mTimeZone);
         }
```

```
@@ -261,42 +260,12 @@ public class ExtendedMessageFormat extends MessageFormat {
      * @return the hashcode
      */
+    @Override
+    public int hashCode() {
+        int result = super.hashCode();
+        result = HASH_SEED * result + ObjectUtils.hashCode(registry);
+        result = HASH_SEED * result + ObjectUtils.hashCode(toPattern);

+    }

@@ -172,7 +172,7 @@ public class Week extends RegularTimePeriod implements Serializable {
     */
    public Week(Date time, TimeZone zone) {
        // defer argument checking...
-       this(time, zone, Locale.getDefault());
+       this(time, RegularTimePeriod.DEFAULT_TIME_ZONE, Locale.getDefault());
    }

@@ -1669,7 +1669,7 @@ public class SystemUtils {
      *
      * @return the version, for example 131 for Java 1.3.1
      */
-    static int toJavaVersionInt(String version) {
+    static float toJavaVersionInt(String version) {
         return toVersionInt(toJavaVersionIntArray(version, JAVA_VERSION_TRIM_SIZE));
     }
```

Recommendations

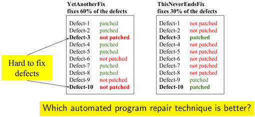
Repair research should evaluate if new techniques repair hard and important defects.



Is the bug important to fix?

Is the bug hard to fix?

Is the patched program correct?



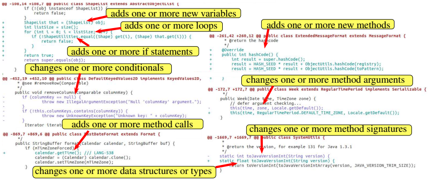
YetAnotherFix fixes 60% of the defects		ThisNeverEndsFix fixes 30% of the defects	
Defect-1	patched	Defect-1	not patched
Defect-2	patched	Defect-2	not patched
Defect-3	**not patched**	**Defect-3**	**patched**
Defect-4	patched	Defect-4	not patched
Defect-5	patched	Defect-5	not patched
Defect-6	not patched	Defect-6	not patched
Defect-7	patched	Defect-7	not patched
Defect-8	patched	Defect-8	not patched
Defect-9	not patched	Defect-9	patched
Defect-10	**not patched**	**Defect-10**	**patched**

Hard to fix defects

Which automated program repair technique is better?

Repair research should target defects that existing techniques have missed.

Evaluation benchmarks need to account for diversity of defect complexity, importance, etc.



```
@@ -108,14 +108,7 @@ public class ShapeList extends AbstractObjectList {
         if (!(obj instanceof ShapeList)) {
             return false;
         }
+        ShapeList that = (ShapeList) obj;
+        int listSize = size();
+        for (int i = 0; i < listSize; i++) {
+            if (!ShapeUtilities.equal((Shape) get(i), (Shape) that.get(i))) {
+                return false;
+            }
+        }
+        return true;
-        return super.equals(obj);
     }

@@ -452,19 +452,10 @@ public class DefaultKeyedValues2D implements KeyedValues2D,
      * @see #removeRow(Comparable)
      */
     public void removeColumn(Comparable columnKey) {
-        if (columnKey == null) {
-            throw new IllegalArgumentException("Null 'columnKey' argument.");
-        }
-        if (!this.columnKeys.contains(columnKey)) {
-            throw new UnknownKeyException("Unknown key: " + columnKey);
-        }
         Iterator iterat

@@ -869,7 +869,6 @@ public class FastDateFormat extends Format {
      */
     public StringBuffer format(Calendar calendar, StringBuffer buf) {
         if (mTimeZoneForced) {
+            calendar.getTime(); /// LANG-538
             calendar = (Calendar) calendar.clone();
             calendar.setTimeZone(mTimeZone);
         }
```

```
@@ -261,42 +260,12 @@ public class ExtendedMessageFormat extends MessageFormat {
      * @return the hashcode
      */
+    @Override
+    public int hashCode() {
+        int result = super.hashCode();
+        result = HASH_SEED * result + ObjectUtils.hashCode(registry);
+        result = HASH_SEED * result + ObjectUtils.hashCode(toPattern);

+    }

@@ -172,7 +172,7 @@ public class Week extends RegularTimePeriod implements Serializable {
     */
    public Week(Date time, TimeZone zone) {
        // defer argument checking...
-       this(time, zone, Locale.getDefault());
+       this(time, RegularTimePeriod.DEFAULT_TIME_ZONE, Locale.getDefault());
    }

@@ -1669,7 +1669,7 @@ public class SystemUtils {
     *
     * @return the version, for example 131 for Java 1.3.1
     */
-   static int toJavaVersionInt(String version) {
+   static float toJavaVersionInt(String version) {
        return toVersionInt(toJavaVersionIntArray(version, JAVA_VERSION_TRIM_SIZE));
    }
```

adds one or more new variables

adds one or more loops

adds one or more if statements

changes one or more conditionals

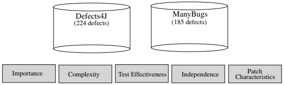
adds one or more method calls

changes one or more data structures or types

adds one or more new methods

changes one or more method arguments

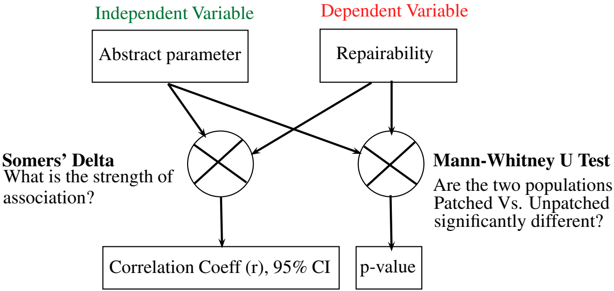
changes one or more method signatures



Annotated datasets and scripts are available at

https://github.com/LASER-UMASS/AutomatedRepairApplicabilityData

Evaluation Methodology



Independent Variable

Abstract parameter

Dependent Variable

Repairability

Somers' Delta
What is the strength of association?

Mann-Whitney U Test
Are the two populations Patched Vs. Unpatched significantly different?

Correlation Coeff (r), 95% CI

p-value